

LAPORAN TUGAS PEMROGRAMAN B KOMPUTASI NUMERIK



Disusun Oleh:

Kelompok 3

Daffa Hardhan - 2306161763

Muhammad Bryan Farras - 2306230975

Tri Yoga Arsyad - 2306161920

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KOMPUTER
UNIVERSITAS INDONESIA
TAHUN AJARAN 2024 / 2025**

BAB I

ABSTRAK

Proyek ini mengaplikasikan metode iterasi titik tetap untuk menyelesaikan persamaan non-linear $\cos(x) = x$. Persamaan ini cukup sulit untuk diselesaikan dengan cara aljabar biasa karena x terintegrasi dalam fungsi trigonometri, sehingga pendekatan numerik menjadi diperlukan. Metode iterasi titik tetap dimulai dengan tebakan awal, diikuti dengan perhitungan berulang nilai x menggunakan rumus $x = \cos(x)$ hingga hasilnya mendekati nilai tetap atau konvergen.

Program ini dikembangkan menggunakan bahasa C untuk mengotomatisasi proses iterasi dan menampilkan hasil setiap langkahnya. Berdasarkan percobaan yang telah kali lakukan, solusi yang ditemukan mendekati akar persamaan adalah sekitar $x \approx 0,739$. Temuan ini mengindikasikan bahwa metode iterasi titik tetap cukup efektif dan mudah diterapkan untuk mencari akar dari persamaan non-linear secara numerik.

BAB II

PENDAHULUAN

Dalam aktivitas sehari-hari dan dalam ilmu pengetahuan, kita sering kali menghadapi persamaan matematis yang sulit dipecahkan secara langsung, contohnya adalah $\cos(x) = x$. Karena sifatnya yang rumit, penyelesaian perlu menggunakan pendekatan numerik yang bisa memberikan hasil secara bertahap.

Metode titik tetap merupakan salah satu teknik numerik yang cukup sederhana dan efisien. Dimulai dari sebuah perkiraan awal, metode ini melakukan iterasi sampai ditemukan hasil yang konsisten. Dalam pemrograman B ini, metode tersebut kami implementasikan menggunakan bahasa pemrograman C untuk lebih memahami cara penyelesaian persamaan non-linear secara praktis.

BAB III

STUDI LITERATUR

Metode iterasi titik tetap merupakan salah satu metode numerik untuk menemukan solusi dari persamaan $f(x) = 0$ dengan cara mengubahnya menjadi format $x = g(x)$. Proses ini dimulai dari suatu perkiraan awal, kemudian dilanjutkan dengan iterasi hingga nilai yang diperoleh tidak berubah (konvergen).

Dalam laporan ini, diterapkan fungsi $g(x) = \cos(x)$, karena fungsi $\cos(x)$ bersifat kontinu dan hasilnya terletak dalam rentang -1 hingga 1, sehingga sangat sesuai untuk pendekatan iteratif ini. Pendekatan ini diimplementasikan menggunakan bahasa C.

BAB IV IMPLEMENTASI DAN HASIL

1.1. Penjelasan Data yang Digunakan

Dalam proyek ini, fungsi $g(x) = \cos(x)$ digunakan sebagai pendekatan untuk persamaan $\cos(x) = x$. Nilai awal x_0 ditentukan sebesar 0,5, yang dipilih secara subjektif selama masih berada dalam jangkauan fungsi $\cos(x)$.

Proses iterasi dilaksanakan secara berkelanjutan hingga perbedaan nilai x menjadi sangat kecil, yakni kurang dari 0.0001, atau sampai mencapai batas maksimal 100 iterasi. Hal ini dilakukan untuk menghindari perulangan yang tidak berujung dan memastikan bahwa proses terhenti saat nilai semakin mendekati akar dari persamaan tersebut.

1.2. Penjelasan Metode yang Digunakan

Metode titik tetap dilaksanakan dengan cara merubah persamaan menjadi $x = g(x)$, di mana dalam hal ini $g(x) = \cos(x)$. Proses iterasi dimulai dari nilai awal x_0 , kemudian dihitung secara berkelanjutan: $x_1 = \cos(x_0)$, $x_2 = \cos(x_1)$, dan seterusnya, sampai perbedaan antara dua iterasi terakhir sangat kecil.

Program C kami buat guna untuk menjalankan prosedur ini secara otomatis. Setiap hasil iterasi akan ditampilkan sampai mencapai ambang toleransi kesalahan yang ditentukan. Pada akhirnya, program ini menunjukkan nilai x yang mendekati solusi dari persamaan $\cos(x) = x$.

1.3. Diskusi dan Analisa Hasil Eksperimen

Hasil dari penelitian menunjukkan bahwa metode titik tetap dapat memberikan hasil yang dekat dengan solusi dari persamaan $\cos(x) = x$ dengan cepat. Dari iterasi pertama hingga yang ke-22, nilai x terus

mengalami perubahan dengan perbedaan yang semakin kecil, hingga akhirnya mencapai konvergensi di sekitar $x \approx 0,739050$.

Fungsi $g(x) = \cos(x)$ terbukti memenuhi kriteria konvergensi karena nilai mutlak dari turunannya kurang dari 1 di sekitar titik akar. Ini memberikan kestabilan serta efisiensi pada metode titik tetap untuk kasus ini. Program dalam bahasa C yang kami kembangkan berhasil menggambarkan proses iterasi dengan jelas dan tepat.

1.4 Output Program

```
[Running] cd "c:\Users\Tri yoga\Documents\PEMEROGGRAMAN B\CODE PROGRAM\" && gcc contoh.c -o contoh && "c:\Users\Tri yoga\Documents\PEMEROGGRAMAN B\CODE PROGRAM\"contoh
Metode Titik Tetap:  $x = \cos(x)$ 
Iterasi      x
1      0.877583
2      0.639013
3      0.802685
4      0.694778
5      0.768196
6      0.719165
7      0.752356
8      0.730081
9      0.745120
10     0.735006
11     0.741827
12     0.737236
13     0.740330
14     0.738246
15     0.739650
16     0.738705
17     0.739341
18     0.738912
19     0.739201
20     0.739007
21     0.739138
22     0.739050

Akar mendekati: 0.739050
```

BAB V KESIMPULAN

Berikut Merupakan Kesimpulan dari pemerograma B Komputasi Numerik di atas:

- Metode titik tetap merupakan teknik numerik yang mudah dan praktis untuk menyelesaikan persamaan non-linear seperti $\cos(x) = x$.
- Pelaksanaan metode ini dalam bahasa C memungkinkan iterasi dilakukan secara otomatis dan dengan efisiensi yang tinggi.
- Dari pengujian yang dilakukan, diperoleh nilai yang hampir sama dengan akar persamaan, yaitu $x \approx 0,739$, setelah melalui 22 kali iterasi.

- Fungsi $\cos(x)$ memenuhi kriteria konvergensi, sehingga metode titik tetap dapat berhasil diterapkan.
- Proyek ini memberikan wawasan lebih dalam mengenai penerapan metode numerik dalam bidang pemrograman dan matematika.

Link Github: <https://github.com/Yogaarsyad/PEMROGRAMAN-B.git>

REFRENSI

Burden, R. L., & Faires, J. D. (2011). Numerical Analysis (9th Edition). Brooks/Cole, Cengage Learning.

[1] "Fixed-point iteration," Wikipedia, Dec. 13, 2023.
https://en.wikipedia.org/wiki/Fixed-point_iteration

[2] Wikipedia Contributors, "Numerical analysis," Wikipedia, Nov. 14, 2019.
https://en.wikipedia.org/wiki/Numerical_analysis